1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	May 1991	Professional Paper

4. TITLE AND SUBTITLE

LEAST-SQUARES LEARNING AND APPROXIMATION OF POSTERIOR PROBABILITIES ON CLASSIFICATION PROBLEMS BY NEURAL NETWORK MODELS

5. FUNDING NUMBERS

PR: EEB2
WU: DN088671
PE: 0602234N

6. AUTHOR(S)

P. A. Shoemaker, M. J. Carlin, R. L. Shimabukuro, C. E. Priebe

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Ocean Systems Center
San Diego, CA 92152–5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Naval Ocean Systems Center
Block Programs
San Diego, CA 92152–5000

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

We consider multilayer neural network models which are applied to stochastic classification problems and are trained with error back propagation methods. Expectations for network outputs are weighted least-squares approximations to posterior probabilities for the classes (Gish 1990; Shoemaker, forthcoming; White 1981). In an empirical study, networks were trained on small benchmark problems with known probability density functions, using training data comprising random samples generated according to those functions. Expected classification accuracy and goodness of fit of network outputs to posterior class probabilities were subsequently evaluated. Classification performance near the Bayes optimum was obtained for each problem, and fits to posterior class probabilities were judged reasonable, with root-mean-expected-square differences between outputs and probabilities below 0.05 seen in individual networks for both problems.

Published in *SPIE*, Vol. 1515, 1991.

14. SUBJECT TERMS	15. NUMBER OF PAGES
electronic devices, components and subsystems VLSI RFLSI	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAME AS REPORT

21a. NAME OF RESPONSIBLE INDIVIDUAL	21b. TELEPHONE *(Include Area Code)*	21c. OFFICE SYMBOL
P. A. Shoemaker	(619) 553-5385	Code 552

LEAST-SQUARES LEARNING AND APPROXIMATION OF POSTERIOR PROBABILITIES ON CLASSIFICATION PROBLEMS BY NEURAL NETWORK MODELS

P.A. Shoemaker, Code 552
M.J. Carlin, Code 663
R.L. Shimabukuro, Code 553
C.E. Priebe, Code 421
Naval Ocean Systems Center
San Diego, CA 92152-5000

ABSTRACT

We consider multilayer neural network models which are applied to stochastic classification problems and are trained with error back propagation methods. Expectations for network outputs are weighted least-squares approximations to posterior probabilities for the classes (Gish 1990; Shoemaker, forthcoming; White 1981). In an empirical study, networks were trained on small benchmark problems with known probability density functions, using training data comprising random samples generated according to those functions. Expected classification accuracy and goodness of fit of network outputs to posterior class probabilities were subsequently evaluated. Classification performance near the Bayes optimum was obtained for each problem, and fits to posterior class probabilities were judged reasonable, with root-mean-expected-square differences between outputs and probabilities below 0.05 seen in individual networks for both problems.

INTRODUCTION

Much of the current interest in neural network models centers on their capabilities for tasks such as classification, which typically rely upon data with inherent uncertainty or randomness, and require some *a posteriori* judgement of likelihood or probability. With this interest, analyses in a statistical context of learning in mapping-type networks are beginning to appear in the literature (Gish 1990; Levin *et al.* 1989; Shoemaker, forthcoming; White 1989). Such studies have closed gaps in the understanding of network performance on stochastic problems, by application of (often well-known) statistical techniques and results to neural network paradigms. We discuss one such result, and some empirical investigations of network learning performance which are evaluated in its context. We consider networks applied to classification tasks, which specify class membership with a simple one-of-N output code, and which are trained according to a least-squares criterion using error back-propagation methods (Rumelhart *et al.* 1986). Training is assumed based upon random samples generated on the network input space according to stationary class probability laws, with the class assignments absolutely determined. Each output element in this scheme codes for an individual class, and if the target output is one when the class is present and zero otherwise, then conditional expectations for the target values are the posterior probabilities for the classes (White 1989). Expectations for network outputs in this case are weighted least-squares approximations to those posterior probabilities (Gish 1990; Shoemaker, forthcoming; White 1981), which justifies interpretation of network outputs as indicating degree of confidence in class membership.

The basic qualitative features of the mappings performed by feedforward neural network models bear upon their suitability for fitting posterior probabilities. The sigmoidal activation functions often associated with neuronal analogues or elements possess limited ranges as do probability functions, and in problems in which class probability densities have monotonic overlapping tails, posterior probability for a class will in fact be sigmoidal over a path in the input space on which its density is increasing and that of its complement is decreasing. (Indeed, the logistic activation function allows exact representation of posterior probability in two-class problems with Gaussian statistics (Horne and Hush 1990)). In general problems, when one expects to fit curved hypersurfaces of level probability in the space of the observed (input) variables, it is necessary to use elements whose net inputs are higher than first order in their individual inputs, or multilayer networks of the common neuronal model with first-order or inner-product form of input.

In our empirical study, multilayer feedforward networks were trained with input data derived from two stochastic

classification problems, and evaluated with regard to both their classification performance and the approximations to posterior class probabilities which were developed during training. Two variants of error back-propagation were used. Joint probability densities were established in closed form beforehand, with training sets comprising random sequences of values generated according to those probability densities. This allowed post-training evaluation of statistical expectations for network performance. In addition, distributions of performance criteria, for several network and training sample sizes, were compiled in one of the problems by training and evaluating large sets of networks.

PROBABILISTIC CLASSIFICATION PROBLEMS

Consider N mutually exclusive classes indexed by subscripts c and d (c, d=1..N), and assume that observations are generated by the classes on a space X of observed variables according to continuous joint probability density functions D_c. In addition, let x represent a vector of the observed variables, which serve as inputs to the network, f_c the function on X which specifies the output of the element which codes for class c, $t_c \in \{0,1\}$ the target output for that element, and $\bar{D}_c = \Sigma_{d \neq c} D_d$ the joint probability density for the complement of class c. Then the expected value of the sum-square error on the network's outputs may be written

$$E[\Sigma_c(t_c-f_c)^2]=\Sigma_c\int_X D_c(x)(1-f_c(x))^2+\bar{D}_c(x)(-f_c(x))^2dx \tag{1}$$

which can be shown (Gish 1990; Shoemaker, forthcoming; White 1981) to reduce to

$$E[\Sigma_c(t_c-f_c)^2]=\Sigma_c\int_X D_c(x)(1-P(c|x))^2+\bar{D}_c(x)(P(c|x))^2dx+\Sigma_c\int_X (D_c(x)+\bar{D}_c(x))(P(c|x)-f_c(x))^2dx \tag{2}$$

where P(c|x) is the conditional probability of class c given an observation x. Each integral in the second summation on the right hand side of (2) is the expected value of the square difference between a network output f_c and the posterior probability for class c. For brevity we refer to this sum as EP:

$$EP=\Sigma_c\int_X (D_c(x)+\bar{D}_c(x))(P(c|x)-f_c(x))^2dx \tag{3}$$

It differs from the expected sum-square error by a constant (the first summation on the right-hand side of (2)), and thus minimization of the expected sum-square error in network outputs corresponds to a (weighted) least-squares fit of those outputs to the posterior probabilities. The constant represents the minimum expected square error possible, achieved when each network output is identically equal to the posterior probability for the corresponding class (except possibly at isolated points) over the entire input space X. Network parameters obtained by minimization of square error summed over a finite training set (as well as over network outputs) are statistically consistent estimators for those which minimize the expected sum-square error (White 1981), and thus the conclusions above hold in the limit of large training sets, with the integrals replaced by sums over the training sets.

Bayes optimum classification is achieved by assigning to each class the region of input space over which its probability density is greater than that of any other (mutually exclusive) class. The expected rate EC_0 at which classification is made correctly is then

$$EC_0=\Sigma_c\int_{X_{D_c}} D_c(x)dx,$$
$$\text{where } X_{D_c} \subset X | D_c(x) > D_d(x) \text{ for all } d \neq c \text{ and all } x \in X_{D_c}, \tag{4}$$

and when classification by the network is based upon selection of the class with the largest corresponding output, the expected rate EC at which the network classifies correctly is

$$EC=\Sigma_c\int_{X_{f_c}} D_c(x)dx,$$
$$\text{where } X_{f_c} \subset X | f_c(x) > f_d(x) \text{ for all } d \neq c \text{ and all } x \in X_{f_c}. \tag{5}$$

With joint probability density functions known *a priori*, we were able to evaluate the expected performance of trained networks in our study. Classification accuracy was quantified using the expected rate of correct classification EC. As a

measure of goodness of fit of network outputs to actual posterior class probabilities, we used the root-mean value of EP, the expected sum-square difference between outputs and posterior probabilities. The mean in this latter measure was taken over the network outputs, and also over sets of networks where so indicated. Values of the performance measures were obtained by numerical integration over a large region of input space containing class means.

Each of the two problems used in the study was defined on a two-dimensional input space, to allow easy visualization of probability densities, decision regions, and other objects of interest. The first problem was fabricated to provide a computationally undemanding benchmark. Three mutually exclusive classes were assigned heavily overlapping, circularly symmetric Gaussian joint probability densities with equal variances and prior probabilities. Mean values of the observed variables for classes 1-3 were (-1/2,1/2), (1/2,1/2), and (1/2,-1/2), respectively, and the standard deviation of each distribution was 1/2. This problem is particularly simple in that Bayes optimum classification can be achieved with a Voronoi partition about the class means. At the Bayes optimum, the expected rate EC_o at which classification is correct is 77.9%.

The second problem used estimated probability densities for a six-class problem based upon two particular characteristics of radar emissions. The density estimates (whose sum is depicted in Fig. 1) were developed using 50-150 data (depending upon prior probability) per class, and for five of the six classes consisted of mixtures of Gaussians with diagonal covariances. The exception was a single Gaussian. All data used in network training trials were derived from these density estimates, rather than the original data set. They were normalized so as to lie within the square $(-1,1)^2$. This problem requires considerably more complex decision boundaries for optimum classification than does the first, although the overlap of the densities is smaller. EC_o for this problem is 96.3%.

SIMULATIONS

The learning algorithms we employed were forms of error back-propagation corresponding to stochastic approximation (White 1989; Robbins and Munro, 1951), in which weight changes are based upon presentation of individual input and target

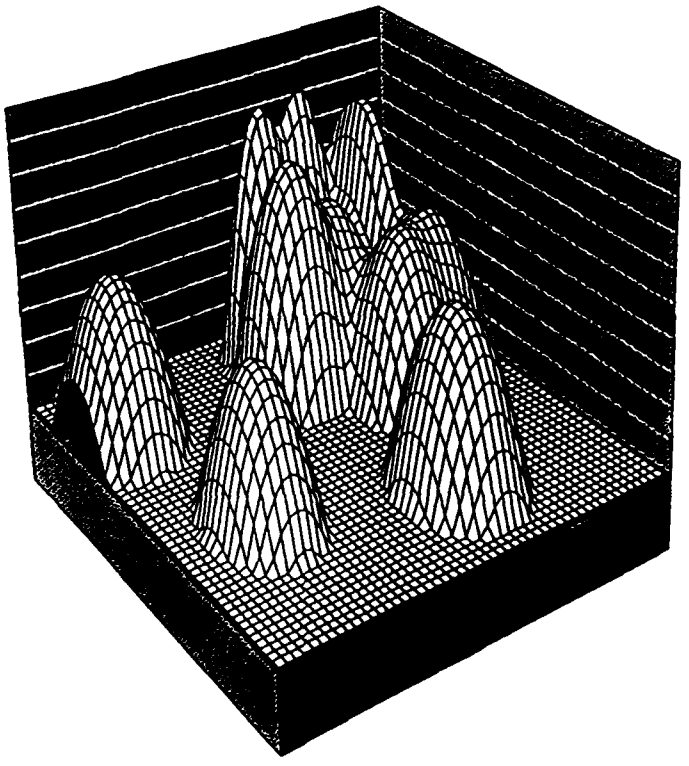

Fig. 1. Estimated total probability density function for the radar classification task, in log scale over 3½ decades.

vectors, without "batching" or other schemes requiring memory. The learning rules were applied iteratively on training samples of finite size. The well-known gradient descent version of back-propagation (Rumelhart *et al.* 1986) was used (without "momentum"), as well as a second variant with trinary quantization of weight updates (Shoemaker *et al.* 1990; this learning rule is described briefly in an appendix to this paper). Results reported herein are primarily for standard back-propagation, with a few comparisons with the trinary version where indicated. The weight update scaling parameter η ("learning rate" for standard back-propagation or increment/decrement magnitude for the trinary version) was in most cases decreased according to a 1/t schedule, where t indexes the number of training data presentations. The precise form used was $\eta = \eta_o/(1+ \lambda t)$, where suitable ranges for the initial value η_o and decay rate λ were determined empirically for each problem. Training sets were generated as random sequences of "observations," in which a class was chosen at random according to its prior probability, and a random pair of input values was generated according to the class probability density function.

The networks which we simulated consisted of elements with first-order (weighted sum plus bias or offset) net inputs. They had single hidden layers, with the number of hidden elements varied for each problem. Two, three, four, and five hidden elements were used for the three-class Gaussian problem, and eight, thirteen, and eighteen for simulations on the radar classification problem. As noted, each network had two inputs, and three outputs for the Gaussian problem, or six outputs for the radar problem. Initial weight sets were random and uniformly distributed on the interval (1/2,-1/2). Rather than an activation function defined on (0,1), the hyperbolic tangent was used to take advantage of the improved convergence properties which obtain with bipolar-valued activations (Stornetta and Huberman 1987). The target values -1 and 1 replaced 0 and 1, respectively, with approximations to posterior probabilities obtained from network outputs by the transformation $1/2(\bullet + 1)$.

For the Gaussian problem, networks of each hidden layer size (two to five elements) were trained on a sequence of training sets of increasing size. For each combination of network and training sample size, multi-start optimization with 50 learning trials was performed, with each trial based upon different initial weights and training set. The learning parameters were assigned the values $\eta_o = 0.1$ and $\lambda = 1/632$, based upon performance in initial learning trials in which they were varied. Network training was terminated at approximately 20,000 data presentations (t = 20,000) regardless of the size of the individual training set (*e.g.*, 63 iterations were performed on sets with 316 "observations", and 20 iterations on sets with 1000 "observations"). This figure was also determined on the basis of initial trials, in which it was found that networks were always very near a steady state in terms of performance after that number of data presentations with the given learning parameters.

On the radar problem, greater effort was required to identify suitable learning parameters and iteration counts. Most learning trials were based upon training sets with 600 data, which is approximately the number of real data used to form the probability density estimates, although some trials used sets of 3000 data to examine the effect of increased sample size. Parameter values $\eta_o = 0.1$, and λ between 1/5000 and 1/1500 were used in most trials. Learning was terminated after 60,000 - 150,000 data presentations. Typical performance figures were calculated over 20 trials for three different hidden layer sizes.

Because the radar problem was based upon realistic data, it was used as a basis for comparison of classification performance with some well-known alternative classifiers. Linear and quadratic classifiers (Duda and Hart 1973), a kernel estimator based upon Gaussians (recently recast in parallel form as a "probabilistic neural network" (Specht 1990)), and a single nearest-neighbor classifier using Euclidean metric were evaluated. A suitable window parameter for the kernel estimator was determined empirically. Fifty runs, each based upon a separate 600-point training sample, were performed for each classifier, and expected performances were calculated. Variance in the performances of these classifiers arises from sampling variance, whereas in the neural network model, additional variance arises due to convergence on various suboptimal weight sets. Therefore, we performed five training runs (each from a different initial weight set) on each of the fifty training samples, and retained the best-performing network of the five, to reduce this second source of variance. Networks had 13 hidden units, with $\eta_o = 0.1$ and $\lambda = 1/5000$, and runs terminated at t=150,000.

RESULTS

On the Gaussian problem, networks of all sizes trained with standard back-propagation proved capable of classification performance very near the Bayes optimum. Networks of each size were observed with EC within 0.1% of this optimum. As might be expected in a three-class problem, though, networks with two hidden elements did not fit posterior probabilities

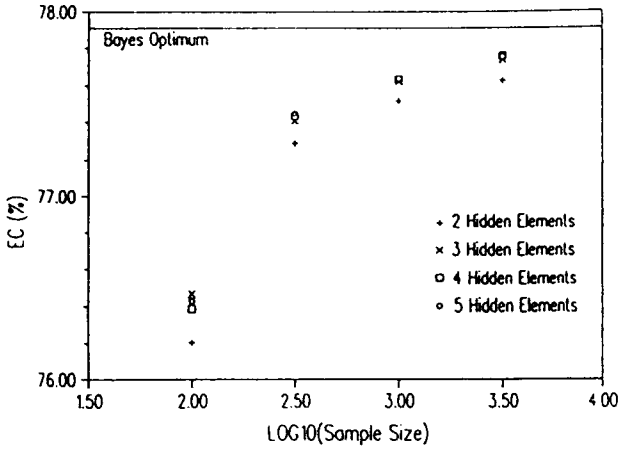

Fig. 2a. Mean values of the expected rate of correct classification plotted against training set size, for networks trained on the Gaussian classification problem.

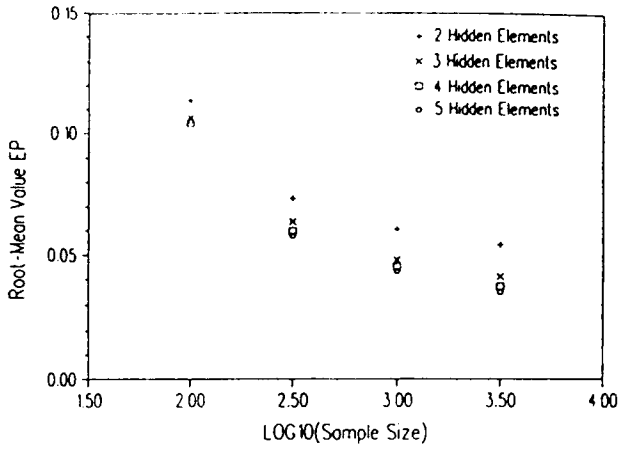

Fig. 2b. Root of the mean expected square difference between network outputs and posterior probabilities vs. training set size. Each mean is taken over fifty networks.

as well as networks with three or more hidden elements; the root-mean value of EP was greater than 0.045 for all networks with two hidden elements, while values between 0.025 and 0.03 were attained by the best networks of each larger size.

Fig. 2a depicts mean values of EC and Fig. 2b depicts root-mean values of EP, with means taken over fifty networks at each hidden layer and training set size. Histograms are given in Fig. 3 for networks with three hidden elements and for each training set size, and demonstrate how variance of each network performance measure decreases with variance in the sampling distributions, as training sample size increases. These trials were performed with the goal of a sieve-like (Grenander 1981) correlation of increases in sample size with allowable increases in network complexity. However, the results indicate that networks with three or more hidden elements are capable of such close approximation to posterior probabilities, that increases in this capability with model complexity (*i.e.*, hidden layer size) are masked by training sample variability (or indeed, by possible variations due to convergence of networks on various local optima in the parameter space with performance very near but varying slightly from the global optimum).

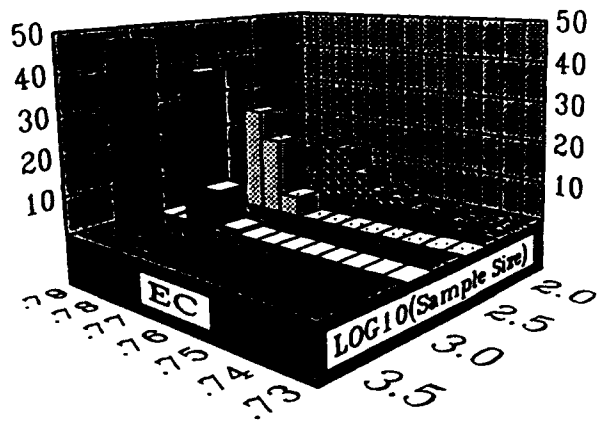

Fig. 3a. Histograms of expected rate of correct classification for networks with three hidden elements, and for several training sample sizes.

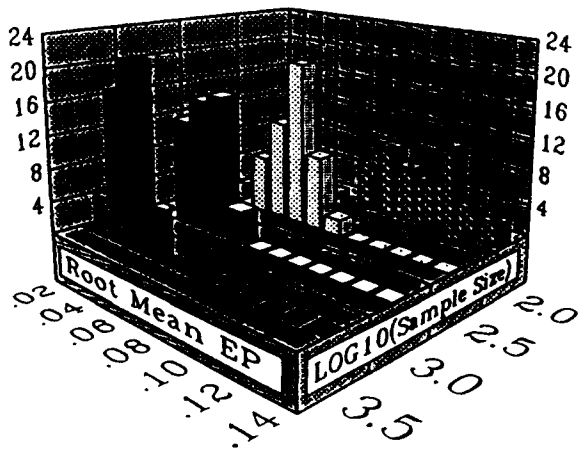

Fig. 3b. Histograms of root mean expected square difference between network outputs and posterior probabilities. Fifty data compose each histogram.

Characteristics of some of the best solutions are shown in Fig. 4. All of these networks achieved expected classification rates of better than 77.8% correct. Fig. 4a shows Bayes optimum decision boundaries and curves on which actual probabilities are 1/2, for comparison with the network decision boundaries and surfaces of zero output (estimated probability = 1/2) in Figs. 4b - 4f. Fig 4b, which is for a network with two hidden elements, shows a somewhat poorer approximation to posterior probabilities than the larger networks (*e.g.*, in a substantial region in the lower left of the graph, predicted probabilities for two classes are both greater than 1/2). The character of the three-hidden-element solution in Fig. 4c is of particular interest. It is straightforward to construct a perceptron with threshold logic elements and a three-element hidden layer, which is capable of performing a Voronoi partition and thus achieving Bayes optimum classification on the Gaussian problem: a hidden element "hyperplane" (*i.e.*,the line in this case on which net input to the element is zero) is placed coincident with each of the optimum decision boundaries, and each output element is configured to compute an intersection of two appropriate half-spaces defined by hidden element responses. One might expect to find a similar solution, with hidden element hyperplanes lying along decision boundaries, in trained three-hidden-element sigmoid networks. However, this is generally not the case; solutions as a rule more nearly resemble that in Fig. 4c. With this configuration, the network uses the analog nature of its activation function to obtain good approximations to posterior probabilities. This illustrates the danger of thinking about the operation of networks with sigmoid activations in terms of the properties of threshold logic unit perceptrons.

Fig. 4f is also for a network with three hidden elements. Its performance is nearly as good as that of the network in Fig. 4c, despite the evident differences in hidden layer hyperplane placement. This illustrates another conclusion resulting from the study: multiple minima (beyond those due to network symmetries) in sum or expected square error apparently do exist for this problem and architecture; or perhaps more likely, entire manifolds exist in parameter space on which performance is near (least-squares) optimal and gradients are very small.

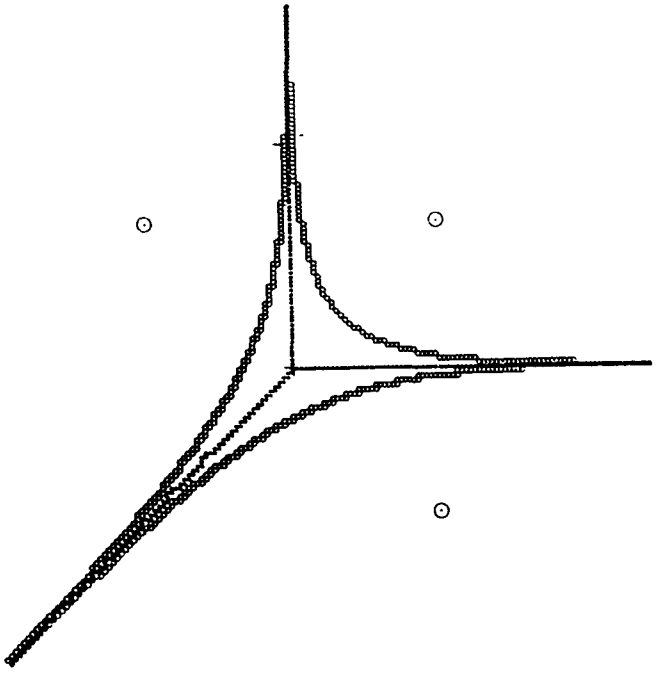

Fig. 4a.

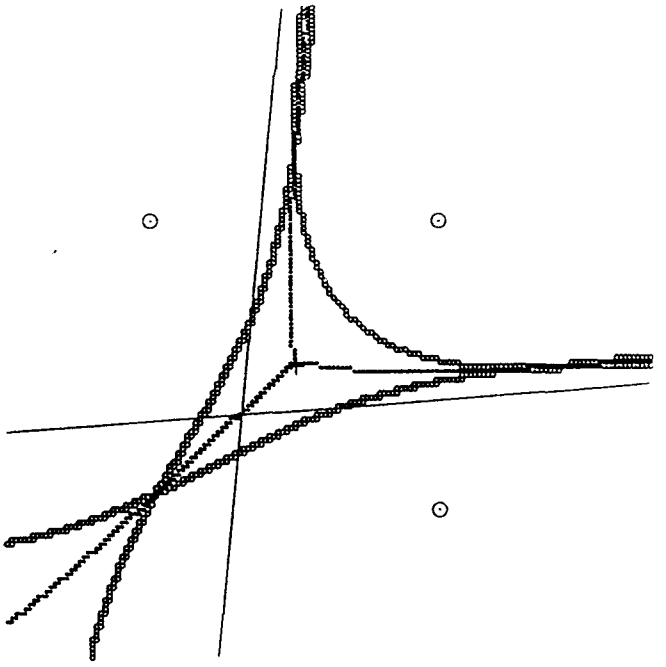

Fig. 4b.

Fig. 4. Characteristics of some solutions to the Gaussian problem found by networks with two to five hidden elements. Large circles with centered dots indicate class means. Open circles indicate points on the curves on which estimated probabilities are 1/2. Filled symbols are points on the decision curves, on which two or more estimated probabilities are equal. Solid lines indicate the position of hidden-layer "hyperplanes", or lines on which net inputs to the hidden layer elements are zero. In (a) are shown actual surfaces on which probabilities are 1/2, and Bayes optimum decision curves, for purposes of comparison. In (b)-(e) are shown results for networks of from two to five hidden elements, respectively. In (f) are results for a second network with three hidden elements, for comparison with (c).

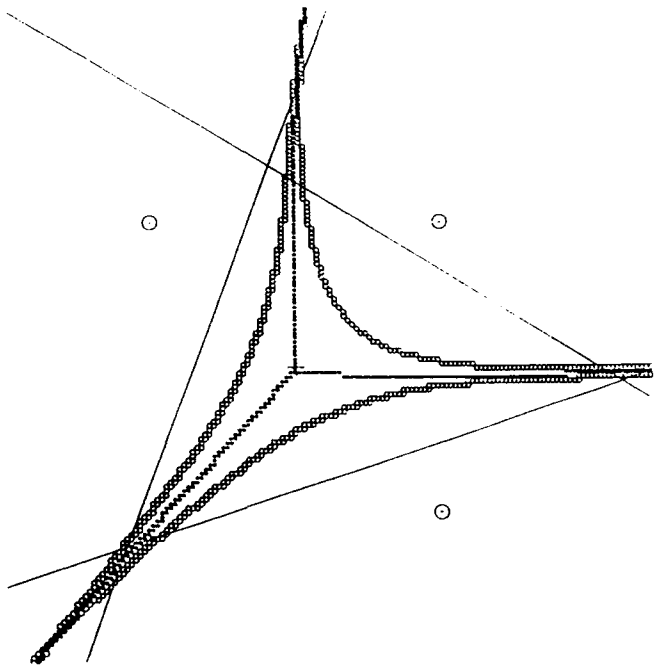

Fig. 4c.

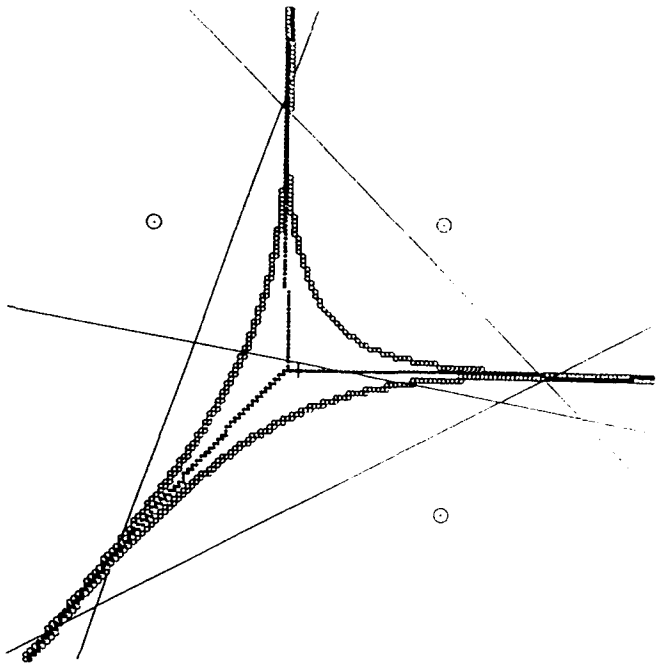

Fig. 4d.

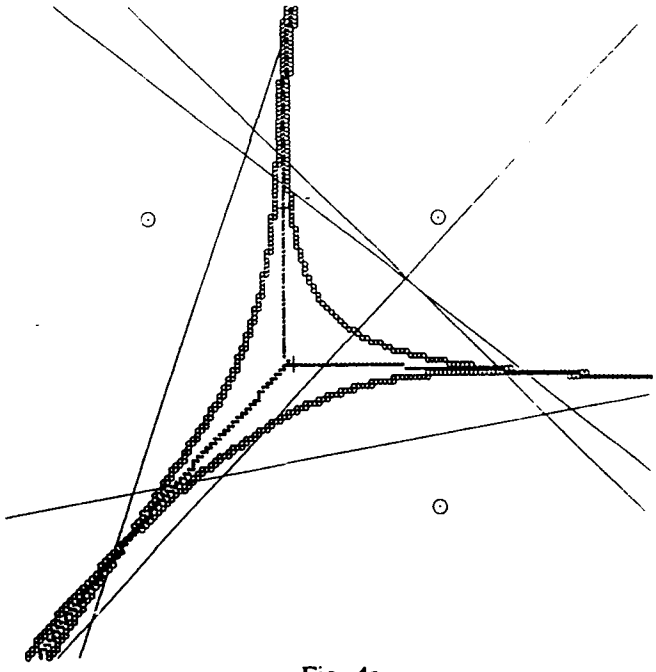

Fig. 4e.

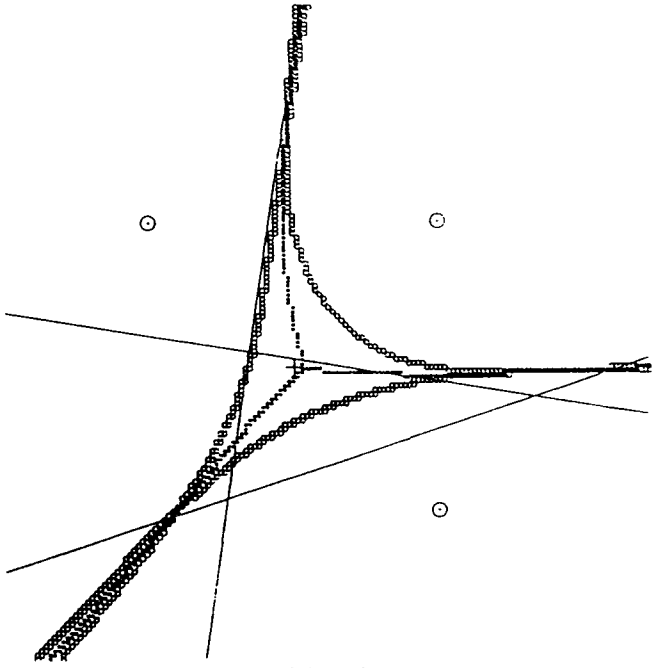

Fig. 4f.

On the radar classification problem, the best networks had EC values of about 95.7%, or 0.6% worse than for Bayes optimum classification. The lowest root-mean values of EP were about 0.046. Mean values of EC and root-mean values of EP were calculated over 20 learning trials with $\eta_0 = 0.1$, $\lambda = 1/5000$, and which were terminated after 150,000 data presentations. For networks trained on data sets containing 600 "observations," these values were respectively 95.3% and .058 for 8 hidden elements; 95.4% and .055 for 13 hidden elements; and 95.4% and 0.054 for 18 hidden elements. For networks with 13 hidden elements trained on samples with 3000 "observations," the values were 95.6% and 0.050, respectively. Fig. 5 shows Bayes optimum decision regions, and decision regions formed by one of the thirteen hidden-element networks. The complex nature of the problem, and the approximations made by the network, are evident.

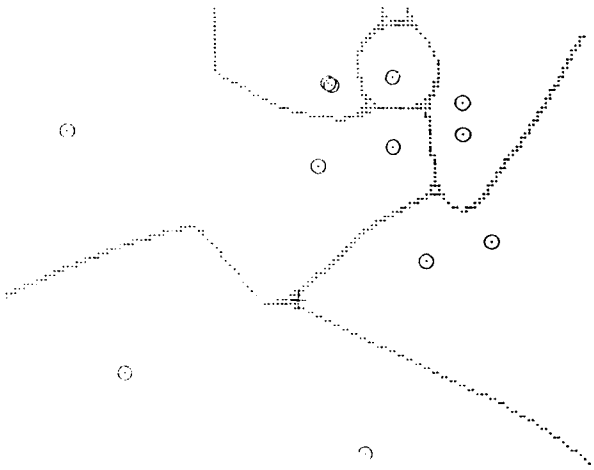

Fig. 5a.

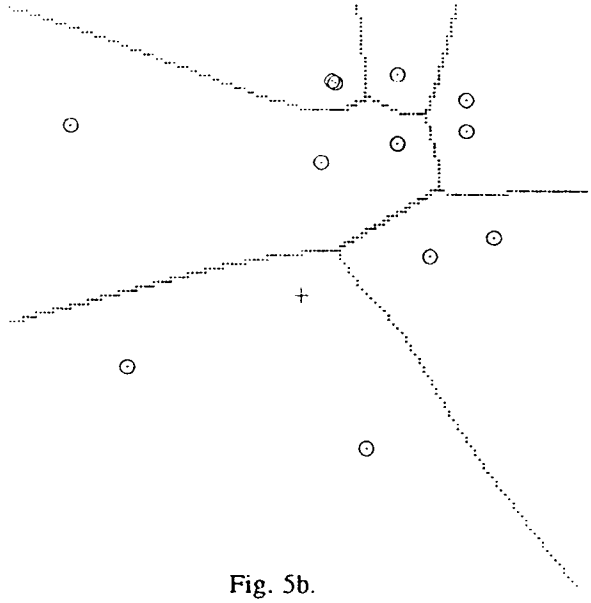

Fig. 5b.

Fig. 5. Classification characteristics of a solution to the radar problem found by a network with thirteen hidden elements. Means of individual Gaussians which compose the probability density functions are indicated by large circles with centered dots. Filled symbols lie on the decision curves, on which two or more estimated probabilities are equal. In (a) are shown Bayes optimum decision curves, for purposes of comparison. In (b) are shown results for the network. Lower left corners of these figures correspond to the corner in the foreground of Fig. 1.

It is worthwhile to recall in viewing Figs. 4 and 5 that network *approximations* are expected to be best in regions of highest probability density.

Results of the comparison with other classifiers is shown below in Table 1.

	LC	QC	KERN	SNN	ANN
EC, % (Mean)	89.1	95.9	95.9	94.2	95.5
(St. Dev.)	0.51	0.13	0.15	0.58	0.22
Root-Mean EP (Mean)	0.180	0.037	0.039	0.099	0.052
(St. Dev.)	0.004	0.004	0.005	0.010	0.005

Table 1. Classification performance of five classifiers. LC = Linear Classifier, QC = Quadratic Classifier, KERN = Kernel Estimator, SNN = Single Nearest Neighbor, ANN = Neural Network Model. Means and standard deviations are taken over fifty trials. Bayes optimum EC is 96.3%.

When we applied the variant of back-propagation with trinary weight updates to the two problems, we could not find ranges of the learning parameters for which its performance after a given number of data presentations was comparable to that of standard back propagation. (This rule has an additional free learning parameter ε_2 which influences the fraction of weights which receive non-zero updates.) The best trinary networks yielded expected performance measures which were consistently slightly worse than those of the best standard back-propagation networks (for instance, values of EC differed by about 0.4% for the radar problem). This situation was somewhat the opposite of that for the deterministic problems we

have studied (Shoemaker *et al.* 1990), in which the trinary rule led to more rapid convergence to suitable solutions. To improve performance, we examined more complicated learning procedures with two regimes, in which learning parameters assumed two different sets of values in two different parts of the learning trial. We obtained best performance by training for a period with little or no decay in the learning constant and with a relatively small ε_2 (allowing many non-zero weight updates), followed by a regime with λ comparable to the values used for standard back-propagation, and with larger ε_2. Classification performance comparable to that of the standard back-propagation networks could be attained in this way, although approximations to posterior probabilities were generally slightly poorer. For example, 20 networks were trained on the radar problem with such a procedure for 150,000 total presentations, using samples with 3000 "observations." A mean EC of 95.5% and root-mean EP of 0.063 were obtained.

CONCLUSIONS

This study was intended to provide some empirical data on the performance of networks trained with back-propagation techniques with regard to both expected classification accuracy and approximation of posterior probabilities on stochastic classification tasks. In two small benchmarks (one based upon circularly symmetric Gaussian probability densities and the other on estimated densities for a radar classification problem), networks trained on sufficiently large training sets by standard back-propagation proved capable of very respectable classification performance, as measured by expected rates of correct classification relative to the Bayes optima. Rates within 0.1% of the Bayes optimum were observed for the Gaussian problem, and within 0.6% of the Bayes optimum for the radar problem. Network outputs provided fairly good approximations to posterior probabilities for the classes, as measured by the root of the mean expected square difference between network outputs and posterior probabilities. This took values as low as 0.025 for the Gaussian problem and 0.046 for the radar problem. The poorer performance on the radar problem reflects its more complicated statistics. However, in a comparison with several other classifiers on this same problem, the networks attained either better or nearly comparable performance. Networks trained with trinary back-propagation (see appendix) generally performed somewhat more poorly than standard back-propagation networks, although a learning procedure was developed which allowed classification with about the same accuracy.

Of course, expected performance cannot be calculated in real-world problems when learning by example is required; however, these results suggest that multilayer sigmoid "neural" networks trained with back-propagation methods can attain good classification performance, when common-sense *ad hoc* methods of choosing network size are used in combination with relatively unbiased performance measures. Further empirical work needs be done; the efficacy of alternative objective functions for training (El-Jaroudi and Makhoul 1990; Movellan 1990) is a promising subject for investigation.

ACKNOWLEDGMENTS

This work was supported by the Office of Naval Technology, Project RS34M40-N02A (Microelectronics Technology). The authors thank W. Huang, D. Marchette, and H. White for helpful comments.

APPENDIX

In the "trinary" form of error back-propagation learning (Shoemaker *et al.* 1990), weight and bias updates are quantized into three states according to the rule

$$\Delta W_{ij} = \begin{cases} \eta\, sgn(\delta_i O_j) & (|O_j| \geq \varepsilon_1 \text{ and } |\delta_i| \geq \varepsilon_2) \\ 0 & (|O_j| < \varepsilon_1 \text{ or } |\delta_i| < \varepsilon_2) \end{cases} \tag{6}$$

$$\Delta B_i = \begin{cases} \eta\, sgn(\delta_i) & (|\delta_i| \geq \varepsilon_2) \\ 0 & (|\delta_i| < \varepsilon_2) \end{cases} \tag{7}$$

where δ_i is the delta term associated with the i^{th} element (Rumelhart *et al.* 1986), O_j is the output of the j^{th} element, W_{ij} is the weight connecting the j^{th} to the i^{th} element, B_i is the adjustable bias or offset of the i^{th} element, and Δ indicates a weight or bias update. η, ε_1 and ε_2 are positive learning parameters. The constants ε_1 and ε_2 define conditions under which no

update occurs for a given weight, or under which either an increment or decrement is made. In all simulations using this rule, ε_2 was set to 0.33, which value permits a non-zero weight update when the output of the element projecting to a weight is outside the central third of its effective range. The initial scaling constant η_0 was typically set to 0.1, and ε_2 was assigned values as small as 0.04 and as large as 0.5.

Simulations with this learning rule have shown it capable of convergence upon solutions to several small deterministic problems, in total iteration counts which compare favorably with those required by standard back-propagation (Shoemaker *et al.* 1990). Performance is generally best when input data are normalized so as to lie within the interval (-1,1).

REFERENCES

Duda, R.O., and P.E. Hart. 1973. *Pattern Classification and Scene Analysis.* John Wiley and Sons, New York.

El-Jaroudi, A., and J. Makhoul. 1990. "A new error criterion for posterior probability estimation with neural nets." Proceedings, *International Joint Conference on Neural Networks* (San Diego, June 1990), IEEE, Piscataway, NJ, vol. 3, 185-192

Gish, H. 1990. "A probabilistic approach to the understanding and training of neural network classifiers." Proceedings, *International Conference on Acoustics, Speech, and Signal Processing* (Albuquerque, NM, April 3-6). IEEE, Piscataway, NJ, vol. 3, 1361-1364.

Grenander, U. 1981. *Abstract Inference.* John Wiley and Sons, New York.

Home, B. and D. Hush. 1990. "On the optimality of the sigmoid perceptron." Proceedings, *International Joint Conference on Neural Networks* (Washington, DC, January 15-19). Lawrence Erlbaum Associates, Hillsdale, NJ, vol. 1, 269-272.

Levin, E., N. Tishby, and S.A. Solla. 1989. "A statistical approach to learning and generalization in layered neural networks." Proceedings, *Second Annual Workshop on Computational Learning Theory* (Santa Cruz, CA, July/August), R. Rivest, D. Haussler, and M.K. Warmuth, Eds. Morgan Kaufmann Publishers, San Mateo, CA, 245-260.

Movellan, J. 1990. "Error functions to improve noise resistance and generalization in backpropagation networks." Proceedings, *International Joint Conference on Neural Networks* (Washington, Jan. 1990), Lawrence Erlbaum Associates, Hillsdale, NJ, vol. 1, 557-560.

Robbins, H., and S. Munro. 1951. "A stochastic approximation method." *Annals of Mathematical Statistics* 22, 400-407.

Rumelhart, D.E., G.E. Hinton, and R.J. Williams. 1986. "Learning internal representations by error propagation." In *Parallel Distributed Processing, Explorations in the Microstructure of Cognition*, D.E. Rumelhart & J.L. McClelland, Eds. MIT Press, Cambridge, MA, vol. 1, 318-362.

Shoemaker, P.A., M.J. Carlin, and R.L. Shimabukuro. 1990. "Back-propagation learning with coarse quantization of weight updates." In Proceedings, *International Joint Conference on Neural Networks* (Washington, DC, January 15-19). Lawrence Erlbaum Associates, Hillsdale, NJ, vol. 1, 573-576.

Shoemaker, P.A. "A note on least-squares learning procedures and classification by neural network models." *IEEE Transactions on Neural Networks*, forthcoming.

Silverman, B.W. 1986. *Density Estimation.* Chapman and Hall, London.

Specht, D. 1990. "Probabilistic neural networks." *Neural Networks* 3, 109-118.

Stornetta, W.S., and B.A. Huberman. 1987. "An improved three-layer, back propagation algorithm." In Proceedings, *IEEE First International Conference on Neural Networks* (San Diego, CA, June 21-24). IEEE, Piscataway, NJ, vol. 2, 637- 643.

White, H. 1981. "Consequences and detection of misspecified nonlinear regression models." *Journal American Statistical Association* 76, 419-433.

White, H. 1989. "Learning in artificial neural networks: a statistical perspective." *Neural Computation* 1, no. 4: 425-464.